REMARKS

Reconsideration of the application is requested in view of the above amendments and the following remarks.

Claim 13 has been cancelled. Claims 1, 2, 8, 12 and 14 have been amended. Changes made to claims 1, 8 and 12 are supported by at least Figures 1 and 4 and the related description of those figures at page 10, lines 7-9 and the paragraph beginning at page 15, line 18. Editorial changes were made to claims 1 and 14. Changes made to the claims by the current amendment are shown in the attached "VERSION WITH MARKINGS TO SHOW CHANGES MADE."

The drawings were objected to for reasons set forth by the Draftsperson in PTO Form 948. The drawings will be corrected in due course.

Claims 1-5, 7-15 and 17 were rejected under 35 U.S.C. ə 102(b) as being anticipated by Bauer et al., "Hardware/Software Co-Simulation in a VHDL-based Test Bench Approach", *Proceedings of the 34th Design Automation Conference*, June 1997, pages 774-779. Applicants respectfully traverse this rejection.

Bauer discloses one method of hardware/software co-simulation. However, Applicants submit that the structure and method disclosed by Bauer for co-simulation of hardware/software is different than that required by claim 1, 8 and 12.

Bauer discloses a test bench and system with a specific hierarchical structure for interfacing with a unit under test; the unit under test being akin the "hardware model" required by claims 1, 8 and 12. The system disclosed by Bauer requires a master synchronizer that uses commands to control applications that in turn interface with a unit under test, as shown in Figure 1 of Bauer. Bauer further discloses in Figure 2 that the microprocessor application, which is controlled by the master synchronizer, includes a message handler, V-ware, a bus functional model that communicates with the unit under test, and a software link that links the microprocessor application to software through a series of send and receive FIFOs, a software server and a software client. The master synchronizer reads a "control file" and writes a "report file" via VHDL text I/O (see Figure 1 of Bauer). The applications stimulate the unit under test

and get the responses (see section 4.1 of Bauer). The master synchronizer and microprocessor application are intended to be reusable for multiple usages (see section 3, paragraph 4).

Bauer fails to disclose a hardware model that includes a CPU bus functional model, and further fails to disclose a network for providing communication between the CPU bus functional model and a CPU server that is in communication with the software, as required by claims 1, 8 and 12. The disclosure in section 5.3 of Bauer is directed the hierarchical structure discussed above that includes a master synchronizer, a microprocessor application that includes the bus functional model, and a unit under test and software which are each coupled to the microprocessor application.

The unit under test disclosed by Bauer fails to include a bus functional model because the bus functional model is integrated into a separate microprocessor application (see Figure 2 of Bauer). Separation of the bus functional model into the microprocessor application is necessary in order to make the test bench reusable for developing/testing different units under test. The microprocessor application also acts as a generator, an analyzer, or a mixture of both for purposes of interfacing with the unit under test (see section 4.4 of Bauer). Thus, by routing the command and control signals from the software through the microprocessor application, the software "acts" as a generator and analyzer for the unit under test, although it is actually the microprocessor application that is the analyzer or generator. Without the microprocessor application and master synchronizer, the unit under test and software disclosed by Bauer would not be able to communicate or co-simulate. Therefore, Applicants submit that Bauer fails to disclose every limitation of claims 1, 8 and 12, and the claims that depend from them. Withdrawal of the rejection is respectfully requested.

Claims 6 and 16 were rejected under 35 U.S.C. ∋ 103(a) as being unpatentable over Bauer in view of Rowson, "Hardware/Software Co-Simulation", *Proceedings of the 31st ACM/IEEE Conference on Design Automation Conference*, June 1994, pages 439-440. Applicants respectfully traverse this rejection.

As discussed above, Bauer fails to disclose every limitation of claims 1 and 12. Rowson fails to remedy the deficiencies of Bauer as it relates to claims 1 and 12. Therefore, Applicants

submit that claims 6 and 16 are allowable for at least the reason they are dependent upon an allowable base claim. Applicants do not concede the correctness of this rejection.

In view of the above, Applicants request reconsideration of the application in the form of a Notice of Allowance.

Respectfully submitted,

MERCHANT & GOULD P.C.
P.O. Box 2903
Minneapolis, Minnesota 55402-0903
(612) 322-5300

Date: 5/27/03

Joshua N. Randall
Reg. No. 50,719
JNR:PSTdb:ae

VERSION WITH MARKINGS TO SHOW CHANGES MADE

In the Claims:

Claim 13 has been cancelled.

Claims 1, 2, 8, 12 and 14 have been amended as follows:

1. (Amended) A method of developing an ASIC[, comprising] using a hardware model, a software, and a network, the hardware model comprising a CPU bus functional model, the software being coupled to a CPU server, and the network configured to provide communication between the hardware model and the software, the method comprising the steps of:

developing the hardware model and the software concurrently; and

[co-simulating] communicating command and control information directly between the CPU server and the CPU bus functional model over the network, and communicating command information directly between the CPU bus functional model and the CPU server over the network to co-simulate the hardware model and the software [therebetween via a network] while the hardware model and the software are being developed.

2. (Amended) The method of claim 1, wherein the hardware model is developed on a workstation.

8. (Amended) A method of co-simulating a hardware model and a software in ASIC development, the hardware model comprising a CPU bus functional model, the software being coupled to a CPU server and communicating with the hardware model via a network coupled to the bus functional model and the CPU server, the method comprising:

requesting an access to [a] the hardware model from [a hardware side to a] the software [side via a] via the network;

invoking a function call by [a] the CPU server [at a software side];

sending an access request from the [hardware side] bus functional model to the CPU server [at the software side] via the network; [and]

routing the access request to the hardware model;

developing the hardware model and the software concurrently; and

co-simulating the hardware model and the software while the hardware model and the software are being developed.

12. (Amended) An apparatus for hardware model and software co-simulation in ASIC development, comprising:

a hardware model [to represent], the hardware model representing a hardware board circuit to be co-simulated/tested, the hardware model being developed on a workstation and including a CPU bus functional model;

a software [to command and control accesses], the software providing command and control access of the hardware model, the software being developed/debugged on a target board concurrently with a design of the hardware model, the target board including a CPU server in communication with the software; and

a network[,] coupled between the [workstation] CPU bus functional model and the [target board,] CPU server to communicate a command from the software to the hardware model and to route contents of the command between the [workstation] hardware model and the [target board] software, thereby providing co-simulation of the hardware model and software.

14. (Amended) The apparatus of claim [13] 12, wherein the software is loaded on the CPU server.